

a)

CONTROL SYSTEM WITH BUS BRIDGE

Patent Number: JP2001344194
Publication date: 2001-12-14
Inventor(s): NAKAYAMA SACHIKO
Applicant(s): OKI ELECTRIC IND CO LTD
Requested Patent: ☐ JP2001344194
Application Number: JP20000161487 20000531
Priority Number(s):
IPC Classification: G06F13/36
EC Classification:
Equivalents:

Abstract

PROBLEM TO BE SOLVED: To provide a control system with a bus bridge capable of increasing a processing efficiency.

SOLUTION: This control system comprises a bus bridge 16 interposed between a system bus 11 having a CPU and a memory connected thereto and an extension bus 12 having an I/O device connected thereto. The system further comprises a control circuit 13 occupying the system bus 11 only in the necessary and minimum period even during the transfer of an DMA and a data buffer 14 for temporarily storing the data read from the memory when reading the DMA. Thus, the system bus access standby time of the CPU can be shortened.

Data supplied from the esp@cenet database - I2

2001-344194

[CLAIMS]

[Claim 1] A control system having a bus bridge between a system bus to which a CPU and a memory are connected and an extension bus to which an I/O device is connected, characterized in that it comprises (a) a control circuit which occupies said system bus only for a minimum but necessary period of time also during DMA transfer, and (b) a data buffer for temporarily storing data read out from said memory upon DMA reading, and (c) said control system allows shortening of system bus access waiting time of said CPU.

[Claim 2] A control system having a bus bridge between a system bus to which a CPU and a memory are connected and an extension bus to which an I/O device is connected, characterized in that (a) it comprises a control circuit for releasing said extension bus before acquisition of said system bus in a DMP writing cycle and executing an extension bus cycle after waiting completion of a system bus cycle upon DMA reading, and (b) said control system allows prevention of deadlock of said system bus and said extension bus.

[Claim 3] A control system having a bus bridge as set forth in claim 1 or 2, characterized in that a circuit capable of identifying whether a DMA requesting signal DRQ inputted from said I/O device to said bus bridge is a DMA read request or a DMA write request is added to said bus bridge.

a)

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開2001-344194

(P2001-344194A)

(43) 公開日 平成13年12月14日 (2001. 12. 14)

(51) Int.Cl.⁷

G 0 6 F 13/36

識別記号

3 1 0

F I

G 0 6 F 13/36

ターム(参考)

3 1 0 F 5 B 0 6 1

審査請求 未請求 請求項の数 3 O L (全 6 頁)

(21) 出願番号 特願2000-161487 (P2000-161487)

(22) 出願日 平成12年 5 月31日 (2000. 5. 31)

(71) 出願人 000000295

沖電気工業株式会社

東京都港区虎ノ門1丁目7番12号

(72) 発明者 中山 祥子

東京都港区虎ノ門1丁目7番12号 沖電気
工業株式会社内

(74) 代理人 100089635

弁理士 清水 守 (外1名)

Fターム(参考) 5B061 BA01 BA03 BB15 DD09 DD11

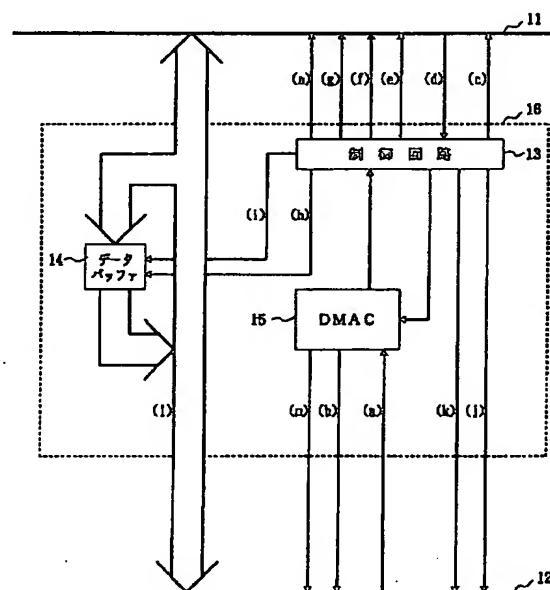
FF07 GG02 GG13 PP02 PP05

(54) 【発明の名称】 バス・ブリッジを有する制御システム

(57) 【要約】

【課題】 処理効率を向上させることができるバス・ブリッジを有する制御システムを提供する。

【解決手段】 CPUとメモリが接続されるシステムバス11とI/Oデバイスが接続される拡張バス12間にバス・ブリッジ16を有する制御システムにおいて、DMA転送中でも必要最小限の期間だけ前記システムバス11を占有する制御回路13と、DMAリード時に前記メモリからリードしてきたデータを一時保管するためのデータバッファ14とを具備し、前記CPUのシステムバスアクセス待ち時間を短縮する。



【特許請求の範囲】

【請求項1】 CPUとメモリが接続されるシステムバスとI/Oデバイスが接続される拡張バス間にバス・ブリッジを有する制御システムにおいて、(a)DMA転送中でも必要最小限の期間だけ前記システムバスを占有する制御回路と、(b)DMAリード時に前記メモリからリードしてきたデータを一時保管するためのデータバッファとを具備し、(c)前記CPUのシステムバスアクセス待ち時間の短縮を可能とすることを特徴とするバス・ブリッジを有する制御システム。

【請求項2】 CPUとメモリが接続されるシステムバスとI/Oデバイスが接続される拡張バス間にバス・ブリッジを有する制御システムにおいて、(a)DMAライトサイクル時においてシステムバス獲得前に前記拡張バスを解放し、DMAリード時においてシステムバスサイクル完了を待ってから拡張バスサイクルを実行する制御回路を備え、(b)前記システムバスと拡張バスのデッドロックの回避を可能とすることを特徴とするバス・ブリッジを有する制御システム。

【請求項3】 請求項1又は2記載のバス・ブリッジを有する制御システムにおいて、前記バス・ブリッジにI/Oデバイスから入力されるDMA要求信号DRQがDMAリード要求なのか、DMAライト要求なのかを識別できる回路を付加したことを特徴とするバス・ブリッジを有する制御システム。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、DMA転送機能を持つI/Oデバイスを有する制御システムに関するものである。

【0002】

【従来の技術】従来、このような制御システムとしては、以下に示すようなものがあった。

【0003】図3はかかる従来の制御システムの構成図である。

【0004】図3において、1はシステムバス9に接続されるCPU(中央演算処理装置)、2はシステムバス9に接続される主記憶メモリ、3はバス・ブリッジ、4、5はシステムバス9に接続されるI/Oデバイス、6～8は拡張バス10に接続されるI/Oデバイスである。

【0005】このように、CPU1と主記憶メモリ2、I/Oデバイス4、5はシステムバス9に接続されている。また、I/Oデバイス6～8は、拡張バス10に接続されている。更に、バス・ブリッジ3はシステムバス9と拡張バス10との両方に接続されている。

【0006】次に、この制御システムの動作について説明する。

【0007】バス・ブリッジ3は、ダイレクト・メモリアクセス・コントローラ(以下、DMACと略す)を有

しており、I/Oデバイス6～8はバス・ブリッジ3を経由して、主記憶メモリ2間とのデータ転送を行う。

【0008】以下、DMA転送の詳細について説明する。

【0009】まず、主記憶メモリ2へのDMAライト(DMA書き込み)すなわち、DMA転送によるI/Oリード(I/O読み出し)の場合について説明する。

【0010】I/Oデバイス6からバス・ブリッジ3へDMA転送要求信号(DRQ)が出力される。DRQを受け取ったバス・ブリッジ3は、I/Oデバイス6へDMA転送応答信号(DAK)を返し、システムバス9のバス占有要求信号(HRQ)を、CPU1へ出力する。HRQを受け取ったCPU1は、バス・ブリッジ3へバス占有応答信号(HACK)を返す。バス・ブリッジ3はHACKを受け取るとシステムバス9を占有したこととなる。そこでバス・ブリッジ3はI/Oデバイス6へI/Oリード信号(IORC)を出力してデータをI/Oデバイス6からリードし、主記憶メモリ2に対して、メモリライト信号を出力してリードしたデータを主記憶メモリ2へライトする。DMA転送が完了すると、バス・ブリッジ3はシステムバス9の占有をやめ、システムバス9を解放する。

【0011】次に、主記憶メモリ2へのDMAリード(DMA読み出し)、すなわち、DMA転送によるI/Oライト(I/O書き込み)の場合について説明する。

【0012】I/Oデバイス6からバス・ブリッジ3へDRQが出力される。DRQを受け取ったバス・ブリッジ3は、システムバス9を占有するためにCPU1へHRQを出力する。HRQを受け取ったCPU1はバス・ブリッジ3へHACKを返す。HACKを受け取ったバス・ブリッジ3はシステムバス9を占有したこととなる。そこで、主記憶メモリ2へメモリリード信号を出力して、主記憶メモリ2よりデータをリードする。同時に、I/Oデバイス6へDAKとI/Oライト信号(IOWC)信号を出力し、主記憶メモリ2よりリードしてきたデータをI/Oデバイス6にライトする。

【0013】この時、バス・ブリッジ3はI/Oデバイス6のDMA転送が完了するまで、システムバス9を占有し、主記憶メモリ2へのリード/ライトを行う。

【0014】例えば、システムバス9がEISAバス(Extended Industry Standard Architecture:パソコン用32ビット拡張バスの規格の一つ)の場合、バスマスタ・サイクルはバスクロックの2CLKが1サイクルである。また、拡張バス10がEISAバスまたはISAバス(Industry Standard Architecture:IBM PCAT互換パソコンの標準バス)の場合でも、DMA転送はリード/ライトともに1サイクルは8CLKである。よって、この場合、システムバス9は拡張バス10のDMA転送の間の8CLK期間、

バス・ブリッジ3に占有されるため、6CLK分何もしない状態でバス・ブリッジ3に占有されることとなる。

【0015】

【発明が解決しようとする課題】上記したように、バス・ブリッジ3はI/Oデバイス6のDMA転送が完了するまでの間、システムバス9を占有し続ける。拡張バス10側で行われているDMA転送が、上記のように1サイクル=8CLKであり、システムバス9側のバスマスタ・サイクルが2CLKであるならば、システムバス9は6CLK分は何も動作をしない状態でバス・ブリッジ3に占有されているわけである。

【0016】もし、CPU1がI/Oデバイス4、5に対してアクセスをしたい状態でも、バス・ブリッジ3がバスを占有しているため、I/Oデバイス4、5へのアクセスはできない。CPU1はI/Oデバイス6がDMA転送を完了し、バス・ブリッジ3がシステムバス9を解放するまで、待たされることとなる。すなわち、主記憶メモリ2とバス・ブリッジ3間のデータ転送時間とは、バス・ブリッジ3とI/Oデバイス6とのDMA転送と同じ時間がかかっているのである。このような状態は、CPU1にとって大変効率が悪い。

【0017】次に、主記憶メモリ2へのDMAライト(I/Oリード)と、CPU1からのI/Oデバイスに対するスレーブ・アクセスとの競合が発生した場合について説明する。

【0018】I/Oデバイス6からバス・ブリッジ3へDMA転送要求信号(DRQ)が出力される。DRQを受け取ったバス・ブリッジ3は、I/Oデバイス6へDMA転送応答信号(DAK)を返し、システムバス9のバス占有要求信号(HRQ)をCPU1へ出力する。HRQを受け取ったCPU1は、バス・ブリッジ3へバス占有応答信号(HACK)を返す。この時、CPU1がI/Oデバイス7、8に対してスレーブ・アクセスをしようとした場合、以下のような状態となる。

【0019】CPU1はバス・ブリッジ3を経由してI/Oデバイス7、8に対してスレーブ・アクセスを行おうとするが、I/Oデバイス6がDMA転送中であり、拡張バス10を使用しているためにDMA転送が終わるまで待ち状態となる。

【0020】一方、バス・ブリッジ3はバス占有要求信号(HRQ)を出力して、システムバス9を占有しようとするが、CPU1がI/Oデバイス7、8に対してスレーブ・アクセス中でバスを占有できないため、スレーブ・アクセスが終わるまで待ち状態となる。すなわち、システムバス9側でも拡張バス10側でも待ち状態となり、バスのデッドロックが発生する。システムの動作は途中で止まったまま動かない状態となる、という問題が発生する。

【0021】本発明は、上記問題点を除去し、処理効率を向上させることができるバス・ブリッジを有する制御

システムを提供することを目的とする。

【0022】

【課題を解決するための手段】本発明は、上記目的を達成するために、

〔1〕CPUとメモリが接続されるシステムバスとI/Oデバイスが接続される拡張バス間にバス・ブリッジを有する制御システムにおいて、DMA転送中でも必要最小限の期間だけ前記システムバスを占有する制御回路と、DMAリード時に前記メモリからリードしてきたデータを一時保管するためのデータバッファとを具備し、前記CPUのシステムバスアクセス待ち時間の短縮を可能とすることを特徴とする。

【0023】〔2〕CPUとメモリが接続されるシステムバスとI/Oデバイスが接続される拡張バス間にバス・ブリッジを有する制御システムにおいて、DMAライトサイクル時においてシステムバス獲得前に前記拡張バスを解放し、DMAリード時においてシステムバスサイクル完了を待ってから拡張バスサイクルを実行する制御回路を備え、前記システムバスと拡張バスのデッドロックの回避を可能とすることを特徴とする。

【0024】〔3〕上記〔1〕又は〔2〕記載のバス・ブリッジを有する制御システムにおいて、前記バス・ブリッジにI/Oデバイスから入力されるDMA要求信号DRQがDMAリード要求なのか、DMAライト要求なのかを識別できる回路を付加したことを特徴とする。

【0025】

【発明の実施の形態】以下、本発明の実施の形態について図を参照しながら詳細に説明する。

【0026】図1は本発明の第1及び第2実施例を示す制御システムの要部構成図である。

【0027】この実施例では、システムバスの占有期間を最小限にするようにしている。

【0028】この図において、11はシステムバス、12は拡張バス、13は制御回路、14はデータバッファ、15はDMAC、16はバス・ブリッジである。

【0029】また、(a)はDMA転送要求信号DRQ、(b)はDMA転送応答信号DAK、(c)はシステムバス11のバス占有要求信号HRQ、(d)はバス占有応答信号HACK、(e)はバスマスタのスタート信号START、(f)はバスマスタのコマンド信号CMD、(g)はバスマスタのメモリアドレス/ライト信号MEM_R/W、(h)はデータバッファ14の出力イネーブル信号Buf fOE、(i)はデータバッファ14のデータラッチ信号Buf f_LCH、(j)は拡張バス12側のI/Oライト信号IOWC、(k)は拡張バス12側のI/Oリード信号IORC、(l)は内部データバス、(m)はDMAの終了通知信号DMA_DONE、(n)はシステムバス側のサイクル/ウェイト信号CYC_WAITである。

【0030】以下、この制御システムの動作について説

明する。

【0031】まず、DMAライト、すなわちDMA転送によるI/Oリードの動作について説明する。

【0032】DMA転送を行う際には、まず、I/OデバイスよりDRQ(a)がバス・ブリッジ16に入力される。入力されたDRQ(a)はバス・ブリッジ16内部のDMAC15に入る。DMAC15は拡張バス12へDAK(b)とDMA DONE(m)を出力してI/Oデバイスからデータをリードすると同時に、DRQ(a)が入力されたことを制御回路13へ通知する。

【0033】すると、制御回路13はHRQ(c)をシステムバス11へ出力する。制御回路13はHACK(d)を受けると、START(e)、CMD(f)、MEMR/W(W)信号(g)を出力する。これによって、I/Oデバイスからリードしたデータは、内部データバス(1)を通して拡張バス12からシステムバス11へ渡される。1サイクルが終了すると、制御回路13は、システムバス11を解放する。そして、再びDRQ(a)が入力されると、HRQ(c)をシステムバス11へ出力し、システムバス11を占有する。つまり制御回路13は、I/OデバイスのDMA転送中において、バスマスタサイクルによるメモリアクセスに必要最小限の期間だけシステムバス11を占有する制御を行う。

【0034】次に、主記憶メモリへのDAMリード、すなわちI/Oライトの動作について説明する。

【0035】DMA転送を行う際には、メモリライト時と同様にまず、I/OデバイスよりDRQ(a)がバス・ブリッジ16に入力される。入力されたDRQ(a)はバス・ブリッジ16内部のDMAC15に入る。DMAC15はDRQ(a)が入力されたことを制御回路13へ通知する。制御回路13はHRQ(c)をシステムバス11へ出力する。

【0036】制御回路13はHACK(d)を受けると、START(e)、CMD(f)、MEMR/W(W)信号(g)を出力し、メモリからのデータをリードする。リードしたデータは、データバッファ14へ一時保管される。この時、制御回路13はデータバッファ14へBuff_LCH(i)を出力する。

【0037】データバッファ14はBuff_LCH(i)でシステムバス11からリードしたデータを保持する。制御回路13はバスマスタによるリード・サイクルが終わるとすぐにシステムバス11を解放する。DMAC15は、制御回路13がバスマスタ・サイクルの通知を受けると、拡張バス12へDAK(b)とDMA DONE(m)を出力し、データバッファ14に一時保管しておいたデータを拡張バス12へ渡す。

【0038】この時、制御回路13はデータバッファ14へBuff_OE(h)を出力する。データバッファ14はBuff_OE(h)を受けると保持していたデータを拡張バス12側へ出力する。これによって、I/O

Oデバイスからリードしたデータは、システムバス11から内部データバス(1)を通して一時データバッファ14に保管された後、拡張バス12へ渡される。

【0039】制御回路13は、再びDRQ(a)が入力されると、HRQ(c)をシステムバス11へ出力し、システムバス11を占有する。つまり、制御回路13は、I/OデバイスのDMA転送中において、バスマスタサイクルによるメモリアクセスに必要最小限の期間だけシステムバス11を占有する制御を行う。

【0040】このように、第1実施例によれば、制御回路13により、I/OデバイスのDMA転送中においてバスマスタサイクルによるメモリアクセスに必要最小限の期間だけ、システムバス11が占有されるため、バスの占有期間が短くなる。CPUはI/Oデバイスへのスレーブ・アクセスをするためにバスが解放される時間が短くなる。バスの解放はDMA転送中でも1マスタサイクル毎に行われるため、DMA転送中でも他のI/Oデバイスに対するスレーブ・アクセスが可能となり、CPUが長時間待ち状態になることはない。

【0041】次に、本発明の第2実施例について説明する。

【0042】この実施例では、システムのデッドロック発生を回避するようにする。

【0043】以下、動作について図1を参照しながら説明する。

【0044】まず、DMAライト(I/Oリード)とCPUのスレーブサイクルが競合した場合について説明する。

【0045】I/OデバイスよりDRQ(a)が入力され、拡張バス12へDAK(b)とIORC(k)を出力してI/Oデバイスからデータをリードしている最中にCPUからのスレーブ・アクセスがあった場合、制御回路13は拡張バス12側のDMAサイクルが終了し拡張バスが解放されるまでCYC_WAIT(n)を出力し、CPUのスレーブ・サイクルをWait(待機)させる。

【0046】次に、DMAリード(I/Oライト)とCPUのスレーブサイクルが競合した場合について説明する。

【0047】I/OデバイスよりDRQ(a)が入力されHRQ(c)を出力中にCPUからのスレーブ・アクセスがあった場合、制御回路13はCPUからのスレーブ・アクセスが完了するまでDAK(b)の出力を抑止して拡張バス12側のDMAサイクルをWait(待機)させる。

【0048】このように第2実施例によれば、制御回路がI/OデバイスのDMA転送時とCPUのスレーブサイクルの調停を行うことにより、システムバス、拡張バスの両バスが待ち状態となってデッドロックが発生することを回避でき、正常にシステムを動作させることがで

きる。

【0049】次に、本発明の第3実施例について説明する。

【0050】この実施例は、第1、第2実施例に「I/OからのDMA要求がライトかリードかを識別する機能」を付加したものである。

【0051】図2は本発明の第3実施例を示す制御システムの要部構成図である。

【0052】図2において、12は拡張バス、13は制御回路、16はバス・ブリッジ、17はDRQセクタ、18はDMAリード/ライト識別レジスタである。

【0053】更に、(a)はDMA転送要求信号DRQ、(1)は内部データバス、(p)はレジスタ情報通知信号REG_INFO、(q)はDMA_R/Wのライト・イネーブル信号REG_WE、(ar)はリード専用DRQ信号DRQ_r、(aw)はライト専用DRQ信号DRQ_wである。

【0054】以下にこの制御システムの動作について説明する。

【0055】CPUはI/OデバイスへDMA転送の起動をかける前にDMAリード/ライト識別レジスタ18へDMA転送情報をライトする。DMAリード/ライト識別レジスタ18は、一度CPUよりDMA転送情報をライトされると再び情報を上書きされない限り情報を保持し続ける。DMAリード/ライト識別レジスタ18には「これから行うDMA転送がリードか、ライトか」と示す情報(1bit、1=R、0=W)が書き込まれる。

【0056】拡張バス12のI/Oデバイスより、DRQ(a)がバス・ブリッジ16内に入力されると、そのDRQ(a)はDRQセクタ17に入力される。DRQセクタ17は入力されたDRQ(a)がDMAリードなのかDMAライトなのかをDMAリード/ライト識別レジスタ18からの情報REG_INFO(p)より識別し、DMAリードであればDRQ_r(ar)、DMAライトであればDRQ_w(aw)を出力する。

【0057】制御回路13は、入力されたDRQ_r(ar)またはDRQ_w(aw)より、このDMA転送がリードであればDMAリード時の制御を、ライトであればライト時の制御を行う。なお、制御の内容は第1実施例1及び第2実施例で示したものと同様である。

【0058】このように、第3実施例によれば、第1、第2実施例に、I/OからのDMA要求がライトかリードかを識別する機能を追加することで、バス・ブリッジ内の制御回路はDMA転送が始まる前に「DRQがDMAリードサイクルなのか、DMAライトサイクルなのか」を識別することができる。制御回路はI/OからのDMA要求がライトかリードかを転送開始前に識別することで、制御回路の内部論理回路をシンプルに構成することができる。

【0059】本発明は、システムバス上にあるDMACを使用してメモリと拡張バス上のI/Oデバイス間のDMA転送を実現する場合においても有効であり、バス・ブリッジの内部にある制御回路を容易に実現することができる。

【0060】なお、本発明は上記実施例に限定されるものではなく、本発明の趣旨に基づいて種々の変形が可能であり、これらを本発明の範囲から排除するものではない。

10 【0061】

【発明の効果】以上、詳細に説明したように、本発明によれば、以下のような効果を奏することができる。

【0062】(A)制御回路により、I/OデバイスのDMA転送においてバスマスタサイクルによるメモリアクセスに必要な最小限の期間だけ、システムバスが占有されるため、バスの占有期間が短くなる。CPUはI/Oデバイスへのスレーブ・アクセスをするためにバスが解放される時間が短くなる。バスの解放はDMA転送中でも1マスタサイクル毎に行われるため、DMA転送中でも他のI/Oデバイスに対するスレーブ・アクセスが可能となり、CPUが長時間待ち状態になることはなくなる。

【0063】(B)制御回路がI/OデバイスのDMA転送時とCPUのスレーブサイクルの調停を行うことにより、システムバス、拡張バスの両バスが待ち状態となってデッドロックが発生することを回避でき、正常にシステムを動作させることができる。

【0064】(C)バス・ブリッジ内の制御回路はDMA転送が始まる前に「DRQがDMAリードサイクルなのか、DMAライトサイクルなのか」を識別することができる。制御回路はI/OからのDMA要求がライトかリードかを転送開始前に識別することで、制御回路の内部論理回路をシンプルに構成することができる。

【図面の簡単な説明】

【図1】本発明の第1及び第2実施例を示す制御システムの要部構成図である。

【図2】本発明の第3実施例を示す制御システムの要部構成図である。

【図3】従来の制御システムの構成図である。

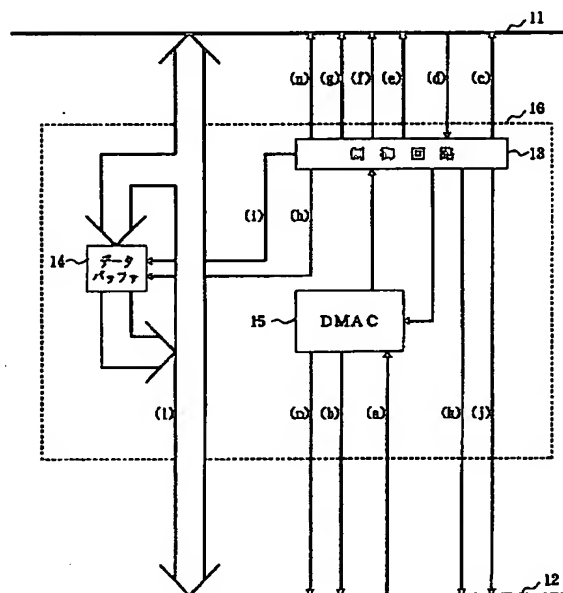
40 【符号の説明】

- 11 システムバス
- 12 拡張バス
- 13 制御回路
- 14 データバッファ
- 15 DMAC
- 16 バス・ブリッジ
- (a) DMA転送要求信号DRQ
- (b) DMA転送応答信号DAK
- (c) システムバスのバス占有要求信号HRQ
- 50 (d) バス占有応答信号HACK

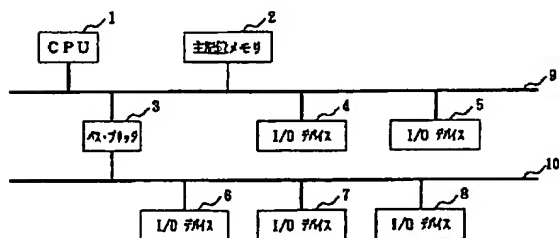
9

- (e) バスマスタのスタート信号START
- (f) バスマスタのコマンド信号CMD
- (g) バスマスタのメモリアド/ライト信号ME
MR/W
- (h) データバッファの出力イネーブル信号Buf
f_OE
- (i) データバッファのデータラッチ信号Buf f
_LCH
- (j) 拡張バス側のI/Oライト信号IOWC
- (k) 拡張バス側のI/Oリード信号IORC
- (l) 内部データバス

【図1】



【図3】



10

- * (m) DMAの終了通知信号DMA_DONE
- (n) システムバス側のサイクル/ウェイト信号C
YC_WAIT
- 17 DRQセクタ
- 18 DMAリード/ライト識別レジスタ
- (p) レジスタ情報通知信号REG_INFO
- (q) DMA_R/Wのライト・イネーブル信号R
EG_WE
- (ar) リード専用DRQ信号DRQ_r
- (aw) ライト専用DRQ信号DRQ_w

*

【図2】

